

かな漢字変換日本語入力システムの 現状と今後に望むこと

青 木 恒 夫

1 はじめに

コンピュータによる日本語処理が日常的になっている現在、日本語入力に対するマン・マシン・インターフェース (man machine interface : MMI) が非常に重要になっている。特に分散処理の主流マシンであるパーソナルコンピュータ (以下、パソコン) における日本語入力システムは著しい発展を見せているが、まだ多くの問題点が存在する。現状の日本語入力システムは日本語ワードプロセッサから派生した経過によって、本来統一的なアーキテクチャ (architecture) が必要な分野であるにもかかわらず、操作の統一や辞書の共有化はなされておらず、ユーザーに多くの負担を強いている。

本稿では1988年9月現在、パソコンにおける主流の日本語入力システムとして広く採用されているかな漢字変換日本語入力システムの現状を分析し、その問題点と共に今後の日本語入力システムに望むことを考察している。

2 日本語入力システム

日本語処理を行うシステムでは、7000種を上回るキャラクタを取り扱わなければならない。これは欧米語の約400種に比較すると、非常に大きなハンディキャップを背負っていることになる。日本語入力システムとは、コンピュータに日本語を快適に、速く、正確に入力することを目的として考案されたソフトウェアシステムである。

2.1 日本語入力システムの呼称

コンピュータに日本語データを入力するためのシステムを本稿では日本語入力システムと呼んでいる。この呼称は各ソフトハウスの商品名や従来からの慣例に従って種々の異なった呼び名が存在する。一般にOSとユーザーの中間に位置する組込型のソフトウェア (デバイスドライバ : device driver) が多いことから、次のような呼称が用いられている。

- 1) 日本語入力プロセッサ (processor)
- 2) 日本語入力フロントプロセッサ (front processor)

略して日本語入力FP, 日本語FP, FP

3) 日本語フロントエンドプロセッサ (front end processor)

略して日本語 F E P, F E P

4) フロントエンドプリプロセッサ (front end pre-processor)

略して F E P P

5) かな漢字変換システム, 日本語変換システム, 自由文変換システム

6) 漢字入力システム, 日本語入力システム

本稿では普遍的に日本語を入力するためのシステムという意味から, 日本語入力システムを採用している。

2. 2 日本語入力システムの発達

1980年代始めからパソコンに漢字ROM (read only memory : 読み出し専用メモリ) が搭載され, 漢字を含む日本語が扱えるようになった。当初, 日本語ワープロソフトによる日本語処理が主流であったが, 7000種以上ある文字をどのように入力するかが最大の関心事であり, 種々の日本語入力方式が考案された。

次に各種日本語入力システムの概説を行う。

2. 2. 1 コード入力方式

最初の日本語入力方式である。一般ユーザーが利用するというよりプログラマーが多く利用した。コンピュータで扱う文字には全てコードが振り分けてある。パソコン用漢字コードでは J I S 系漢字コード (旧, 新, NEC) と M S 漢字コード (Microsoft kanji encoding scheme : シフト J I S) が有名である。文字とコードを対比したコード表を片手に, 4桁のコードを1文字分ごと入力していく。非常に時間のかかる作業であり, 日本語を扱うソフトウェアが増えたこと, 開発用ソフトウェアが日本語化されてきたことなどにより, 実用的な日本語入力システムとは言えなくなった。

2. 2. 2 多段シフト方式

報道関係などの日本語ワープロに長年使われてきた方式である。漢字テレタイプライタを原型にしており, 1つのキーに12個の文字を持つ文字群キーと文字群中の必要文字を選択するシフトキーの2段階キー操作で文字を選択する。右手で文字群キーを操作し, 左手で目的の文字位置を示すシフトキーを操作して文字を特定する。特別の訓練を受けたオペレーターを要する入力方式と言える。

2. 2. 3 タブレット方式

ペンタッチ方式とも呼ばれ, 使用する文字の全部 (2500~3000種の漢字, ひらがな, 数字など)

をタブレット上に配置し、目的の文字をタッチペンで押すことにより入力する方式である。初期の日本語ワープロなどに利用された。和文タイプライタと同様に50音順に漢字が配置されているなど初心者にはなじみやすいが、慣れないうちは目的の文字を探すのが大変で、音読みによる単漢字入力になるため効率も良くない。

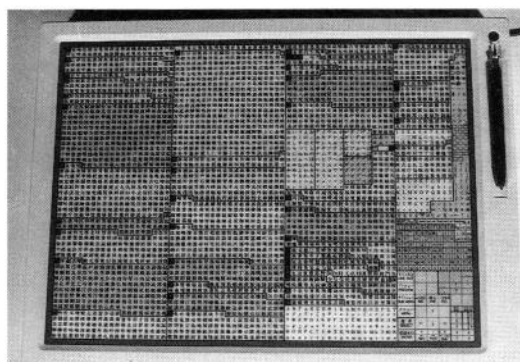


図 2. 1 タブレット

2. 2. 4 2ストローク方式

連想方式とも呼ばれる。通常のキーボードを使用するが、すべての文字を2つのキーで符号化している。「玉」(タマ)は「〇〇」(まるまる)、「力」(チカラ)は「リキ」、「止」(シ)は「ヨス」などと、文字から連想する2つのキーに割り当ててある。この方式は慣れると非常に速く、熟練者は1分間に100~160字程度の文字入力が可能とされている。これもまた専門のオペレーターを要する日本語入力方法で、ワープロオペレーターなどが利用する。

2. 2. 5 かな漢字変換方式

現在、最も一般的な日本語入力システムとなっている。キーボードから入力されたかなを漢字を含む文に変換していく方式である。

初期の頃の漢字入力は**単漢字変換**と呼ばれ、漢字1文字の音読みをかな入力し、変換キーを押すたびに目的の漢字を表示するシステムであった。同音異義語の選択は順に表示される候補文字から選択するか、一覧表の中から選択するようになっている。変換には簡単な辞書が使用される。かな入力の代用として**ローマ字かな変換**による**ローマ字かな入力**が採用され、英文タイプライターに慣れたユーザーには重宝がられた。

その後、辞書の内容も充実し、**熟語変換**が可能になった。熟語変換を更に発展させ、**複合語変換**が行われるようになる。複合語変換では複数の熟語の切れ目を探すための簡単な**文法解析**が必要になる。文法解析が進むと、**文節**を含んだ文を変換する**単文節変換**が可能になる。

また、複数の文節を含んだ**連文節変換**(複文節変換)も行われてきた。連文節変換においては、文法解析がいかにより優れているかが変換効率の善し悪しを決定する。**一文節最長一致法**、**二文節最長一致法**などの文法解析手法が用いられているが、最近では、**AI** (artificial intelligence : 人工知能) 技術を駆使し、意味をも考慮した文法解析も行われている。

また、ユーザーのキーストローク数を減少し、負担を少なくするための**全文一括変換**や**逐次変換**も利用されている。全文一括変換はユーザーが入力するかな文(最大600文字程度)を一気に変換する方式である。一度に正しく変換されることはないが、ユーザーは文節の区切りを修正する

ことだけで正しい日本語に修正でき、文節ごとに変換キーを押す煩わしさから開放される。

一方、逐次変換では、ユーザーがキー入力を始めると、文法解析と同時に辞書の先読みが始まり、適当な文節を発見すると、逐次、日本語に置き換えていく。慣れないうちは戸惑うが、文法解析が正しければ、変換操作をまったく意識しない日本語入力ができる。

その他、「,..」などの句読点や「スペース」(空白)を変換行動の起点とするものもあり、自然な入力を可能にしている。

2. 2. 6 手書き入力方式

1975年頃から電々公社(現NTT)などで研究されており、1, 2のメーカーが商品化している。小型タブレットの上面に貼った薄いアクリル板に専用ペンで文字を書くと、文字認識プログラムが内蔵辞書と照合し、手書きされた文字の文字コードを発生する入力方式である。初期の頃は筆順などが問題になったが、次第に改善され現在に至っている。初心者には非常に好評だが、慣れるに従って入力速度の遅さが目立ち、認識率も完全な実用域には達していない。現状ではキーボードとの併用が主流である。

2. 2. 7 音声入力方式

1980年代始めから実験的な試みがなされ、商品化もされている。最初はDPマッチング方式(dynamic programming method)^{*1}と呼ばれ、特定の標準音声と入力された音声を比較し、単語もしくは単音として捉えるものであった。その意味では、キーボードを音声に置き換えただけであり、意味を理解しての自然な話し言葉入力ではない。その後、意味を考慮した文法解析機能を有したシステムが開発され、次第に実用化の域に達しているようだ。これもAI技術により飛躍的な進歩をみせるであろう。

2. 3 現状の日本語入力システム

以上、多くの日本語入力システムが実用に供されてきたが、現在主流の16ビットパソコンに搭載されているOS、MS-DOS(Microsoft disk operating system: 米Microsoft社登録商標)上では、かな漢字変換方式の利用が圧倒的に多い。これは、特別な付加装置を必要とせず、デバイスドライバと呼ばれるソフトウェアのみでOSの環境の一部として日本語を扱え、初心者にもなじみやすく、他のソフトウェアとの汎用の整合性が得やすいという利便性からきている。

3 かな漢字変換方式の現状

かな漢字変換方式は、より快適な日本語入力をめざし、種々の工夫がなされ現在に至っている。ここでは、現状のかな漢字変換に採用されている単文節変換方式、連文節変換方式、連文節逐次変換方式の各変換方式について述べる。

3. 1 単文節変換方式

2. 2. 5でも述べたが、かな漢字変換方式にも種々の発達段階がある。最も進んでいると言われている逐次変換も、基本的には単文節変換（文節変換）を基礎としている。

文節とは与えられた文を可能な限り多くの意味を持つ最小の単位に区切った時の1単位を言う。子供に向かって“……はネ”とか“……だったネ”などと「ネ」を付けて話すが、この「ネ」を付けて意味の分かる最小の単位が文節である。

テーブルの上に紙と鉛筆が置いてある てーぶるのネ／うえにネ／かみとネ／えんぴつがネ／おいてネ／あるネ

表3. 1 文節の分解

文節は自立語と付属語（助詞，助動詞）からなるのが一般的であるが，文節変換では文節ごとの区切りはユーザーが**変換キー**を押して指示するため，文節の中で最長の自立語を優先的に選択するための解析手法（最長一致法）が一般的に用いられる。

えとは → 干支は OR 絵とは

表3. 2 自立語の違いによる文節の違い

表3. 2の例は，干支はと絵とはの2種類の文節が考えられる。最長一致法を採用した場合，最初に干支は（えと／は）が選択される。前後の意味から絵とは（え／とは）が正しければ，ユーザーが自身で文節内部の切り直しを行い，正しく絵とはに変更する。

最長一致法により常時干支はが選択されては不都合な面が多いので，通常，辞書学習処理によりユーザーの選択結果を保存し，次回の変換時に利用している。また，自立語は接頭辞や接尾辞を伴うことがあり，そのための処理（接辞処理）が必要である他，自立語内部の品詞の種類による付属語の選択作業も必要となる。

現時点では，次に述べる**連文節変換の自動文節区切り**が正常に行われにくいような文の場合にも，ユーザーが文節区切り指示をするため，最も信頼のおける変換方式であり，連文節変換実行中にも多用されている。

3. 2 連文節変換方式

単文節変換方式では，ユーザーが文節の区切りを考えて変換動作を指示するため，コンピュータの文法解析による負担は少ない。連文節変換方式はユーザーの変換指示により，“複数”の文節からなるかな文を適当な文節に分解し，それぞれの文節をかな漢字文に変換していく方式である。この変換方式で問題になるのは，適切な文節に分解する作業であり，正確に行うのは難しい。現状では，文の先頭から最も長い文節を抽出する**一文節最長一致法**と2組の文節の組み合わせが最長になるように文節を抽出する**二文節最長一致法**が一般的に用いられている。

3. 2. 1 一文節最長一致法

1) きょうは／あめが／ふる (今日は雨が降る)
今日は 雨が 降る
2) さくらの花は／ながさ／いた (桜の花が咲いた)
桜のは 長さ 居た
3) だいひょうと／りし／まり／やくしゃ／ちょう (代表取締役社長)
代表と 利子 鞠 役者 長
4) きでは／こを／つくる (木で箱を作る)
木では 子を 作る

表3. 3 一文節最長一致法による文節変換の例

表3. 3は、一文節最長一致法による変換の例である。一文節最長一致法では、先頭から最も長い文節を区切っていくため、最初の文節区切りを間違ってしまうと、最後までおかしい文節が続く。先頭の文節区切りを短く区切り直すと、正常に変換できることが多い。

ユーザーが指示した“文節区切り”を学習保存することは、単文節変換方式ほど単純ではない。それは、ユーザーの意図と自立語の種類により付属語が多様に変化し、文節自身が自在に伸び縮みするためである。単純に結果だけを保存する場合にも大量の情報が必要であり、プロセスも複雑である。現状では、複数回、同一の文節区切りが使用された場合などに、単語使用頻度学習と併用した簡単な文節学習を行っている。

3. 2. 2 二文節最長一致法

3. 2. 1でも述べたが、一文節最長一致法では、最初の文節区切りを間違えると最後まで間違った文節が続くことが多い。一般に1つめの文節が長すぎるため、2つめの文節は非常に短いものになりやすい。そこで、先頭から2個の文節を抽出し、しかも2個の文節の長さが最も長くなるような文節区切りを採用する二文節最長一致法が用いられるようになった。

表3. 4は、二文節最長一致法による変換の好例を示している。第1ステップでは、3種類の文節区切りが考えられる。最初の代表と利子は、一文節最長一致法の文節区切り(最初の文節が最も長くなるよう区切る)にあたる。3つの中で2番目の代表取締役が2文節の合計が最も長く、この段階で代表が固定される。続いての第2ステップでは、固定された代表より後の文が、再度、

第1ステップ	だいひょうとりしまりやくしゃちょう	
	代表と 利子↑	(一文節最長一致法の区切り方)
	代表 取締役 ↑	(二文節合計が最も長い：採用)
	代表と ↑	
第2ステップ	とりしまりやくしゃちょう	
	取締役 社長 ↑	(二文節合計が最も長い：採用)
	取締りや 区↑	
	取締まり 役者 ↑	
	鳥 閉り↑	

表3. 4 二文節最長一致法による変換の例①

切り直しされる。この例では、最初の取締役社長が最も長い文節長となり、最終的に正しく代表取締役社長と変換できる。

第1ステップ	さくらのはながさいた
	桜のは 長さ↑ (二文節の合計が最も長い)
	桜の 花が↑
	桜 野↑

表3.5 二文節最長一致法による変換の例②

表3.5を見てみる。この例では、1番目の桜のは長さが選択されることになる。意味的にはおかしいが、文法的な不都合はない。二文節最長一致法では、このような例が、同じ長さの読みが異なる文節の場合を含め、かなりの割合で頻出する。文節切り直し操作に手間取ると、結果的には一文節最長一致法の方がスムーズな変換結果を得られる場合が多い。

実際には、あまり長い文を一気に変換しようとせず、2文節程度ごとに変換操作を行った方が効率的である。また、文節区切りの学習も行われるので、単語の使用頻度の関係からも次第に変換効率が上がり、使用感は良くなっていく。

3.3 連文節逐次変換方式

逐次変換方式は、現在のところ最も進歩したかな漢字変換の1つである。ユーザーがかな文を入力し始めると、自動的に文節区切りを行い、かな漢字文に変換していく。ユーザーは変換キーを押すことなく、変換結果の正しさだけに神経を集中すればよい。自動文節区切りが正しく行われれば、かなりの確率で変換されるため、改めてコンピュータの“偉さ”を思い知らされる。実際、かなり使い込んでいくと、文節区切り学習や単語使用頻度学習の効果があらわれ、スムーズな変換が可能になる。使い始めの段階では思いどおりの変換結果は得にくいですが、間違った変換が行われた段階で正しい文節区切りの学習を繰り返すと、連文節変換方式同様、次第に効果が上がっていく。

逐次変換方式を含めた連文節変換方式において、操作性を向上させるには、文節区切り直しの操作性向上と高速辞書メディアの採用が必要である。連文節変換方式には必ず文節区切り直し操作が伴う。その操作性が悪いと、単文節変換方式より始末が悪い。特に逐次変換方式の場合、ユーザーの意志とは“無関係”に文節区切り、かな漢字変換が自動的に行われるため、その影響が大きい。

また、逐次変換方式はもちろん、最近の連

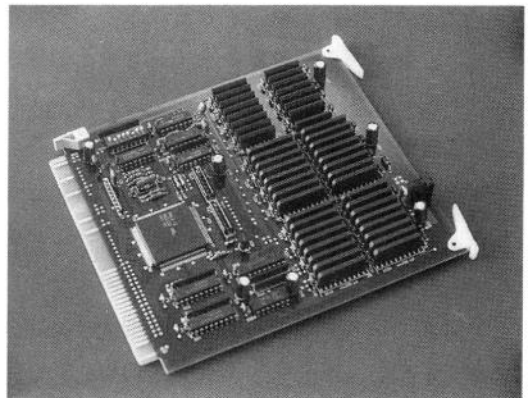


図3.1 RAMディスクボード

文節変換では、変換スピードを上げるため、かな文の入力が始まると、即時、辞書の先読みが行われている。さらに、文法解析が複雑なため、辞書へのアクセス頻度も高い。辞書記録メディアがフロッピーなどの媒体だと、辞書読みにかかなりの時間を要し、スムーズな変換の“足かせ”になってしまう。RAMディスク(ram disk:半導体メモリによる仮想ディスク装置)やハードディスクなどの高速記録メディアの使用が望ましい。

4 日本語入力システムの試用及び比較

かな漢字変換の進歩は激しく、技術的な面だけを捉えれば、専用ワードプロセッサの日本語入力部分を凌ぐ。

本稿の執筆にあたり、多くの市販システム*2を試用し、“汎用”の日本語入力システムとしての役割を考えてみた。試用の準備として100項目の日本語入力システム比較項目を試作し、各システムを比較する指針とした。この日本語入力システム比較項目は、必要機器・環境(ハードウェア・OSなど)、組み込み(インストール:install)、辞書、入力、その他の5項目に大分類し、それぞれの大項目に必要なと思われる比較項目を実際の使用感や各システムの仕様を参考にして作成したものである。以下、この大項目に準じて特筆点を報告する。

4. 1 必要機器・環境

比較項目

- [1-01] 稼働に必要なフロッピーディスクの形式及び最少のドライブ数は
- [1-02] 稼働に必要な最低のメモリ量は
- [1-03] 表示に必要なディスプレイ装置の種類は
- [1-04] ハードディスクを使用できるか
- [1-05] RAMディスクを使用できるか
- [1-06] 辞書ROMを使用しているか
- [1-07] 追記型を含む光ディスクを使用できるか
- [1-08] OS各バージョンとの相性は良いか

最低限のハードウェアと拡張システムについて検討している。日本語入力システムは一般のアプリケーション・プログラム(application program:応用プログラム)とは異なり、言わば影の存在として、できる限り少ないハードウェア資源を使用することが望ましい。高度な機能により年々大型化しているが、パソコンで使用することを前提にした設計が必要である。

ディスプレイ装置について、変換文字の確定と未確定の区別を色分け表示により行っているシステムが多い。単色ディスプレイに対応して、輝度の違いやアンダーラインなどをサポートしているものは問題ないが、カラーディスプレイのみを指定しているものがあるので注意が必要だ。

MGR（メリードレックス）のように辞書をROMの形で供給しているシステムがある。辞書アクセスが高速なため変換速度も速い。ただ、新規の単語登録や辞書学習をフロッピーなどの媒体で行うと、速度の低減は免れない。

追記型を含む光ディスクシステムが実用化されている。5.25インチの追記型タイプでは、1ディスク、最高800MByteのものが市販されており、CD-ROM(compact disk)では、640MByteまで再生できる。意味を考慮した文法解析に必要な大型の辞書ファイルにも対処できるだろう。

4. 2 組み込み

比較項目

- [2-01] インストール専用のプログラムは付属しているか
- [2-02] 任意のドライブ、ディレクトリへシステム及び辞書を設置できるか
- [2-03] システムの数は
- [2-04] システムの大きさは
- [2-05] パラメータの登録法は
- [2-06] 変換方式の変更は可能か
- [2-07] 入力モードの変換は可能か

コンピュータを一部のマニアや専門家が占有する時代は過ぎ、いわば“素人”と呼ばれる一般ユーザーの利用が急速に増加している。マニュアルを読んだだけで、OSに密着した環境設定ファイルや機器の設定を行えるユーザーは非常に少ない。複雑なパラメータの設定などを自動的に行うプログラムが付属しているか否かを日本語入力システムの性能の一部と考えてもよいだろう。実際には、ほとんどのシステムに環境登録用のプログラムが付属していたが、全てのオプションに対応していないものが複数存在した。

任意のドライブやディレクトリへシステム及び辞書ファイルを移動できることは、ハードディスクなどの大容量記録装置を使用する場合、特に重要である。MS-DOSはバージョン2.0の時代から階層ディレクトリをサポートしており、節度あるファイル管理には欠かせない。試用したシステムの中には、VJEのようにシステム及び辞書の所在をルートディレクトリのみ限定しているものがあったが、早急な対策が望まれる。

システム名	システムサイズ(Byte)/個数	辞書サイズ(Byte)/個数	システム・辞書サイズ合計(Byte)
NECTIC	108,880/2	520,192/1	629,072
VJE-Σ	60,621/1	395,264/1	455,885
ATOK 6	75,643/2	454,144/1	529,787
松茸V 2	43,904/1	492,674/3	536,578

表4. 1 日本語入力システムのシステム及び辞書のサイズと数

機能の高度化にともなうシステムや辞書の巨大化が見られる。表4. 1は最新の日本語入力システム (NECTIK, VJE-Σ, ATOK6, 松茸V2) のシステム及び辞書の個数, サイズを示している。MS-DOSのバージョン3. 10 (NEC, PS98-011, 87-10-23) の場合, 1 MByte のフロッピーディスクのフォーマットと同時にシステムを移動すると, 全ディスク空間が1,250,304Byteになる。そのうちシステムプログラム (msdos. sys, io. sys, command. com の3つ) が占める空間が101,985Byteであり, 実質的に残り1,148,319Byteがフリーの領域となる。ところが日本語入力システムのシステムと辞書が1つのフロッピーに共存すると, フリー領域の実に最大55%を占有することになり, 他の大きなアプリケーション・プログラムとの同居が難しくなる。そのため辞書ファイルのみを別のフロッピーに移動するなどの苦肉の策が講じられているが, 機動性や操作性などに問題を残している。現実的には, 既にフロッピーシステムによる稼働が限界にきていることを考え, 大容量のハードディスクやRAMディスクシステム及び辞書を移動して解決するのが望ましい。

変換方式変更は一部のシステム (松茸V2など) で実現しているが, 単・連文節変換や逐次変換方式を起動時に選択したり, 起動中に変更できることである。同様に入力モード変更はかな入力やローマ字かな入力などを自由に選択, 変更できるもので, ほとんどのシステムが可能であった。

4. 3 辞書

比較項目

- [3-01] 辞書の数は
- [3-02] 辞書のサイズは
- [3-03] 辞書登録語数は
- [3-04] 辞書バッファの大きさはユーザーが指定, 変更できるか
- [3-05] 起動中に辞書の変更は可能か
- [3-06] 辞書サイズの拡大縮小はできるか
- [3-07] 辞書の学習は行われるか
- [3-08] 辞書の学習動作を禁止できるか
- [3-09] 文節区切りの学習を行うか
- [3-10] 辞書及び文節学習結果の保存は自動的に行われるか
- [3-11] 単語登録はできるか
- [3-12] 単語登録の方法は
- [3-13] 登録単語の最大長さは
- [3-14] 単語登録時の「読みの長さ」は最大何文字までか
- [3-15] 登録品詞の種類と数は
- [3-16] 単語自動登録は行なわれるか

- [3-17] 単語一括登録はできるか
- [3-18] 英数字による略語登録は可能か
- [3-19] 単語削除の方法は
- [3-20] 読みによる単語一覧削除はできるか
- [3-21] 辞書登録内容を表示できるか
- [3-22] 登録・削除等がユーザーに開放されているか
- [3-23] 辞書の構造・仕様等の情報は公開されているか
- [3-24] 第三者が利用している同一システムの辞書の内容を併合できるか
- [3-25] 辞書メンテナンス用のユーティリティは付属しているか
- [3-26] 他の日本語入力システムと辞書の互換性はあるか
- [3-27] 他の日本語入力システムの辞書を併合できるか

辞書登録語数は公開していないシステムが多い。記録形式が異なるため単純にファイルの大きさだけでは比較できない。最初、辞書登録語数の多さが変換効率に大きく影響すると考え、実際に登録語数の公開されているシステム（ATOK6：47,000語、松茸V2：62,000語）を比べてみた。しかし、文法解析や登録単語の内容（種類や並び）による影響の方が大きいようで、専門用語を多用するような場合を除いて、通常の記事では影響が少ないように思われた。

辞書のファイル名を自由に設定できることは、目的別の辞書を作成し、使用目的に応じた辞書を選択できる利点がある。専門用語を網羅したり、5.4で述べる表記の限定などに利用できる。

単語登録は各システムとも良く似ている。NECTICのように登録時の品詞分類が異常に多いものには閉口した。松茸V2は、通常、少数の基本品詞のみで登録するが、パラメータを変更することにより厳密な活用品詞まで登録できるようになっている。

単語一括登録は、あらかじめ読み、単語、種類 改行 という1単語1行のリストを登録単語数分並べた単語ファイルを作成しておく。次に専用のユーティリティで辞書化し新辞書を作成する。この新辞書を元の辞書と併合すれば、迅速な単語登録が可能になる。データベースなどから出力されたデータ（氏名など）を大量に登録するのに便利だが、採用しているシステムは少ない。

辞書の項目で最も強く感じたのは、各システムごと、全く異なる辞書を使用しているため、辞書の共有が非常に難しかったことである。特別のユーティリティ（市販品）を使用すれば、相互に辞書を利用（併合）できるシステム（ATOK, VJE, 松茸）もあったが、最初からメーカーが準備したものではない。辞書の内部構造はダンプリストを見れば分かるが、各システム、同一形式なのは一つもなかった。メーカーごとの“特色”が必要なのも理解できないことはないが、ユーザーの利益をもう少し考えた辞書作りが必要である。

4. 4 入力

比較項目

- [4-01] 日本語入力システムの起動・終了法は
- [4-02] 変換方式の種類は
- [4-03] 一度に変換できる最大文字数は
- [4-04] 辞書の先読みは行われるか
- [4-05] 辞書の先読みは中止できるか
- [4-06] 変換開始動作は何により行われるか
- [4-07] 文節区切り位置の変更操作法は
- [4-08] 変換対象文節の移動法は
- [4-09] 仮変換文節の確定法は
- [4-10] 仮変換文節の部分確定法は
- [4-11] 変換対象文節の先頭にある助詞の確定法は
- [4-12] 変換可能な入力文字の字種は
- [4-13] 変換動作のキャンセルはできるか
- [4-14] 次候補一覧を表示するか
- [4-15] 次候補一覧からの選択法は
- [4-16] 次候補一覧表示時のカーソル位置は
- [4-17] 単漢字の一覧を表示できるか
- [4-18] 単漢字一覧の最大表示文字数は
- [4-19] 単漢字一覧からの選択法は
- [4-20] 一時的な複合語の変換方法は
- [4-21] コード入力は出来るか
- [4-22] コード入力に使用するコードの種類と数は
- [4-23] コード表を表示できるか
- [4-24] コード表からの入力はできるか
- [4-25] 画面上の特定文字のコードを表示できるか
- [4-26] キートップ上にない記号類の入力方法は
- [4-27] 部首による入力は可能か
- [4-28] 郵便番号による地名入力は可能か
- [4-29] 電話番号による地名入力は可能か
- [4-30] キーボード上の半角文字の直接入力は可能か
- [4-31] キーボード上の全角文字の直接入力は可能か
- [4-32] 入力する文字の字種変更は可能か

- [4-33] 入力した文字の字種変更は可能か
- [4-34] ローマ字かな漢字変換は可能か
- [4-35] 「ん」の入力法は変更可能か
- [4-36] 句読点の変更は可能か
- [4-37] 変換確定文字を元の入力状態に戻せるか
- [4-38] システムラインからの間接入力可能か
- [4-39] システム起動中に変換方法を変更可能か
- [4-40] 変換効率テストの結果はどうか

入力の項目についてはユーザー各自の“慣れ”や“好み”による評価も考慮する必要があった。個人差もあるだろうが、全く新しいシステムを使用した場合、覚えるまでに2時間以上必要だった。これも一応、一通りの入力ではできるといった程度で、複雑な操作に慣れるには、さらに数時間必要であり、日頃使用しているシステムに戻るとホッとしたものである。

操作面では、できるだけキーストローク数が少ない方がよいようである。例えば、日本語入力システムの起動は殆どが **CTRL** キーを押しながら **XFER** キーを押すようになっていく。ところが、松茸V2では同じ動作を **XFER** キーのみのトグル動作で行えるようになっており、操作が非常にやりやすくなった。また、ATOK6の文節切り直し作業は、**←**（左矢印キー）と **→**（右矢印キー）のみで簡単に行なえ、**SHIFT** キーなどと併用した他のシステムに比べ抜群に操作性が良く思えた。3. 3でも述べたように、連文節変換では必ず文節切り直し作業が伴う。切り直し作業自身は単純な作業だが、切り直し1回ごとのキーストローク数の差や操作性の差はシステム全体の操作性に大きく影響することに注意しなければならない。

複数の候補を持つ同音単語の選択方法にも問題があった。**XFER** キーを押しながら、1単語ずつ表示させ選択するもの、候補単語を最下行（システムラインと呼ぶものが多い）に並べられるだけ並べ、カーソルを移動して選択するもの、候補一覧から番号で選択するものと多様であるが、その際に使用するキーがシステムごとに異なるため、混乱が激しく覚えにくかった。

変換効率テストは文章の質に大きく影響され、公平な評価は難しい。完全な評価文章が公開されている訳でもなく、実際の使用感に頼るところが大きいようである。

4. 5 その他

比較項目

- [5-01] システムが占有するメモリ数は
- [5-02] 他のアプリケーションプログラムとの相性は
- [5-03] ファンクションキーの役割を表示してあるか
- [5-04] 確定・未確定文字の表示色を変更できるか

- [5—05] システムラインの表示を消せるか
- [5—06] システムラインの表示色を変更できるか
- [5—07] 外字（利用者定義文字）を利用できるか
- [5—08] 外字を印刷できるか
- [5—09] 利用できる外字の文字数は
- [5—10] 外字編集用のユーティリティはあるか
- [5—11] コピー&ペースト機能はあるか
- [5—12] キー・センス・アイドルの変更は可能か
- [5—13] パソコン自体の実行速度に影響するか
- [5—14] 他のアプリケーションからシステムの起動ができるか
- [5—15] コントロールキーの設定が変更できるか
- [5—16] バージョンの表示ができるか
- [5—17] 単独販売されているか
- [5—18] メンテナンス用のユーティリティは付属しているか

表 4. 2 を見てみる。4. 2 の検討に使用した日本語入力システムをインストールした場合のメモリ使用状況をMS-DOS 単体稼働の場合と比べたものである。確認には全メモリ655,360 Byte の状況で、MS-DOS Ver. 3.10 付属の CHKDSK コマンド (chkdsk. exe, 87-10-23, 9760Byte) を使用した。なお、日本語入力システムが使用する入出力領域として、コンフィギュレーション・ファイル (configuration file, CONFIG. SYS : 環境設定ファイル) には、files=20, buffers=20を記載し、各日本語入力システムにはオプション類のパラメータ (parameter) は付けていない。

MS-DOS 単体の場合の全メモリ量から使用可能メモリ量を差し引いた150,768Byte は、MS-DOS システム及び CHKDSK コマンドが使用するメモリ量とほぼ等しいと考えられる。各日本語入力システムが実際に使用しているメモリ量は、MS-DOS 単体の使用可能メモリ量から各システム使用時の使用可能メモリ量を引くことにより求められる。各日本語入力システムは最低でも約60KByte のメモリを消費する。8086CPU のメモリ領域のほぼ1セグメント分にあたり、数年前の8ビットパソコンのアクセス可能な最高メモリ量 (64KByte) に匹敵する。このこ

システム名	使用可能なメモリ量(Byte)	日本語入力システムが占有するメモリ量(Byte)
MS-DOS単体	565,472	0
NECTIK	459,776	105,696
VJE-Σ	504,592	60,880
ATOK 6	500,128	65,344
松茸V 2	481,264	84,208

表 4. 2 日本語入力システムが占有するメモリ量

とは、アクセスメモリが実質的に640KByteに限定されているMS-DOS Ver. 3.10までのシステムでは、大きな負担になることは否めない。

ほとんどのシステムが外字（利用者定義文字）の扱いを可能としている。ところが画面には表示できるが、システム単体では印刷できないといったシステム（松茸V2など）がある。人名や地名には特殊な文字が多く、データベースなどの利用には必要不可欠なものである。各プリンタとのインターフェースの問題もあるだろうが、解決されたい課題である。

表4.3は、日本語入力システムを使用することによる、パソコン本体の実行速度の変化を調べたものである。測定には、NEC PC980IVX4(CPU80286, クロック周波数8MHz)のフロッピーシステムを使用した。MS-DOSのCOMMAND.COMのダンプリストを3連結したテキストファイル(358,179Byte, 4545行)をTYPEコマンドにより画面に表示した場合の所用時間を内部クロックにより測定している。なお、CONFIG.SYSには前回同様、files=20, buffers=20を記載しており、各システムとも計測中6回のファイルアクセスを認めた。

システム名	所用時間(秒)	MS-DOS単体を100とした時の速度比(%)
MS-DOS単体	162	100
NECTIK	163	99.4
VJE-Σ	182	89.0
ATOK 6	187	86.6
松茸V2	173	93.6

表4.3 日本語入力システムによるパソコン実行速度への影響

1つの日本語入力システムのみを使用している場合は気付かないが、MS-DOS単体で使用した場合に比べて、10%以上速度が低下しているものがある。この結果は当然ながらアプリケーション・プログラムを使用した場合にも見られる。近年のパソコンは高速になったとはいえ、速度低下の因子は少ない方がよい。システムごとに低下割合が違うことから、改善の余地はあると思われる。

4.6 試用を終えて

現在の日本語入力システムは“独立独歩”の性格が非常に強く、**操作の共通性**、**辞書の共有性**、**情報の公開性**、どれを取っても満足のいくものではなかった。特に文節の切り直し方法と辞書の構造は各社各様であり、一つの日本語入力システムを使用していると、別システムへの切り替えが躊躇された。特筆される文法解析技術が確立されていない現在、辞書の内容、特に単語の**登録順**及び**登録単語の質(種類)**が変換効率に大きく影響し、直接、使用感の善し悪しとなる。1ユーザーとして“良いところ取り”にはなるが、最低限、辞書の公開およびマージ(merge: 併合)の必要性を感じた。

また、試用段階で、一部のアプリケーションプログラムとの同時使用ができないシステムが存在した。特に、エディター(editor)使用中にカーソルが消えてしまったり、カーソルの移動がで

きなくなるなどのトラブルを起こすシステム (NECTIK と㈱ピーシーワールドジャパンの ez.com との同時使用) があったが、他アプリケーションとの相性を考慮したオープンな体制での開発が必要と思われる。

5 今後のかな漢字変換システム

かな漢字変換日本語入力システムは完成されたシステムではなく、日々進歩を繰り返している。ここでは、現状の問題点を踏まえ、今後の日本語入力システムに望むことを考察してみた。

5.1 使用頻度による単語学習

現状の単語学習 (辞書学習) は、ユーザーにより選択された単語を単純に同音単語列の先頭に移動し、最終的に使用頻度が高い単語が候補群の前方に集りやすいという性質を利用したものである。ところが、たまたま使用頻度の小さい同音単語を1度でも使用すると、単語列の先頭に移動し、不具合を生じる。また、使用頻度の比較的高い先頭の複数の単語が頻繁に順位を入れ替えるため、逆に使いづらくなることもある。今の問題点は、1回限りの使用が順位を大きく狂わせてしまうことにある。それを解決するには、単語の選択回数を記録させ、複数回 (3~7回程度) 選択された場合のみ順位を入れ替えるような方法を採用すれば良い。ディスクスペースなどの問題もあるので、日本語入力システムの起動中にメモリ記録し、必要回数の使用が認められた場合のみディスクに書き込んだ方が効率的だろう。また、順位の入替えも、一気に先頭まで移動するのではなく、対象単語の一つ前方と入れ替える程度が、長い目で見れば使用頻度の信頼性は高くなる。

5.2 意味解析と文節学習

現段階での変換効率をさらに向上させるには、**意味解析**と効果的な**文節学習**が必要である。現状では、使用されているハードウェア自体の機能の差が大きく、統一的なシステム導入は難しい。しかし、ここ数年、正確には本年 (1988年) が、32ビットパソコンの実質的な幕開けとなっているので、過去の経過から見て、5年程度で実用的なシステムが登場するものと思われる。

現状の連文節変換方式の最大の欠点は、文節区切り作業と使用単語選択に誤りが多いことである。従来、これらの作業は統計的な手法を用いて、最も頻繁に利用される選択肢を選択することにより無難に済ませてきた。この点は "使い込む" ことにより変換効率が向上することから理解

- | |
|--|
| 1) 枯葉返答千円で入院した (彼は扁桃腺炎で入院した)
2) 須も股も股もものうち (李も桃も桃の内)
3) 庭の橋に立つ (庭の端に立つ)
4) 君の石は弱い (君の意志は弱い) |
|--|

表 5.1 誤変換の例 () 内は、意図した文

できる。

ところが前後関係から正しい文節を選択しているわけではないので、思わぬ単語が出現し、苦笑させられることがあった。表5.1の例は、筆者が日頃使用しているかな漢字変換システム(松茸V2)を使用した執筆時点での第1回目の変換結果である。1)の枯葉は辞書内の優先順位により選択されており、返答千円は扁桃腺炎という語句が登録されていなかったために、2文節に分割された結果である。2)も1)同様、李の字句が未登録であり、正しい桃の選択もなされていない。3)及び4)については、意図からは反するが前後関係を無視すれば意味的に正しい。仮に、庭の箸に立つや君の縊死は弱いなどと変換されれば、特別な場合を除き、誤りは明らかとなる。従来の文法解析では、この程度の誤変換は“つき物”であり、上の4例とも、未登録語を登録し、一度正しい文を表示させた後は、学習効果により正しい変換動作が確認されている。

意味解析及び文節学習を行う変換効率向上には2つの方法があると考えられる。

第1は現在の連文節変換方式をもとに、文節前後に使用される語句の妥当性を審査し、最適な文節を選択するという方法である。箸は持ったり投げたりすることはできるだろうが、特別な大きな箸でなければ、“一寸法師”や“蟻”のような体でない限り立つことはできない。また、枯葉は入院することはないだろうが、彼は入院することもある。このように、“語句”は“ある種の語句”とは結びつきにくいという関係を持っているが、この結びつきの程度を解析、学習させることにより、変換効率を上げることができる。

第2は、パターン解析である。我々が外国語を学習する時、文法を覚えるだけでなく、いわゆる“言い回し”の反復練習をし、実際場面では、単語を適当に変形したり、入れ替えたりして利用している。これは一種、構文解析による言語理解の分野になるが、そもそも“意味”の理解が必要な音声認識分野や機械翻訳分野などで盛んに研究されており*3、かな漢字変換における意味解析にも応用できる。AIに於ける自然言語理解システムのパターン認識には、知識を必要とするパターン理解が必須のものとなっているが、膨大な知識と推論を効率良く処理するフレーム理論*4などの優れた理論により、次第に現実的なものとなっている。

将来、キーボードから“かな文”を入力するのではなく、マイクロホンに向かって自然に話しかけると、意図したとおりの“かな漢字文”が画面に表示される日が来るのも決してSF映画に登場する対話システムの中だけの空想物語ではなくなるだろう。

いずれの方法にしろ、膨大な情報が必要になる。人間の脳全体の記憶容量は 10^{12} (一兆)~ 10^{16} (一京)ビットと言われ、その中で本能などの基本的な部分を除いて、意識として記憶できる容量は約100億ビットである。超大型コンピュータの中には全記憶容量100億ビットなどというものも発表されているが、人間のそれに比べれば100万分の1にすぎない。

脳が文節の意味を理解し(考えて)、適切な語句を選択する程度の知識や推論機構を得ることの可否は難しい問題だが、少しでもその働きを模倣した意味解析が行われる可能性は高い。

5.3 日本語入力システムを包含したOS

特定の日本語入力システムがアプリケーションプログラムと相性が悪く、異常な動作を起こすことは、4.6でも述べた。これは、日本語入力システムのデバイスドライバがOSとアプリケーションとの中間的な位置を占めているにもかかわらず、双方、特にアプリケーション側の動作を考慮していないために起こる。日本語入力システムはOSともアプリケーションとも、言わば“肌を接した”関係で相互連絡を行っている。この際、未連絡ポートや共用ポートが存在すると、相互連絡不十分やハングアップの症状を起こす原因となる。

このような問題を解決するため、OSのカーネル(kernel:核)やシェル(shell:外殻)に日本語処理システムを包含したシステムが開発されている。初めての純国産OSと言われるTRO N (the realtime operating-system nucleus)^{*5}では、最初からその考えが凝集されている。また、プログラム開発環境に優れ、多くのワークステーションに採用されているUNIX^{*6}では、カーネル部分に日本語処理システムを組み込めるようになった。

日本語処理システムをOSの一部として扱うことによるメリットは大きい。第1に、各システムごとに異なっている入力操作法の統一が可能となり、ユーザーの負担が非常に軽くなる。

第2に、辞書の共有及びマージが可能になる。辞書は、ユーザーが“育てていくもの”という概念があるが、1ユーザーごと辞書の内容はさまざまである。変換効率の面から言えば、使用頻度の高い単語が登録されているほど結果は良くなる。また、専門用語などには一般性を欠き、全てのユーザーが温存する必要もない。そこで、目的別の辞書を利用したり、同一グループユーザーが登録内容を併合することにより、使いやすいシステムの構築が可能になるだろう。

第3は、プログラム開発労力及び期間が短縮される。最初から日本語処理システムが準備されており、しかもコントロール法が統一されているので、ハードウェアを直接操作するようなことがなければ、新規にシステムを開発したり、各社のデバイスドライバとの相性を心配することもなく、OSの書式に従ったプログラミングを行うだけで、プログラムの安全性が保証される。

OS包含型の日本語入力システムにも欠点がないわけではない。コンピュータの技術開発は非常に速く、5.2でも述べたが、意味解析を行う優秀なシステムが出現する可能性は高い。こういった時に、デバイスドライバのみを変更するわけにもいかず、OS自体を切り替えなければならなくなり、時代の波に乗り遅れやすい。また、非常に使いやすいシステムであれば問題ないが、使いづらいものであれば、致命的な欠陥となってしまう、ユーザーは最悪の環境を強いられることになる。しかし、上記のような欠点が克服されれば、統一された操作法、辞書の共有性というメリットは非常に大きく、今後の主流になる可能性は高い。

5.4 表記を限定できるシステム

日本語ワープロで作成された文章を見ていると、難解な漢字に出会うことがある。日本工業規格のC6226 1983年にはJIS第1水準漢字が2965、第2水準漢字が3388、合計6353文字の漢字

が規定されている。これらの中で漢字使用の目安とされる**常用漢字表**（昭和56年10月1日内閣告示第1号）内の漢字（1945字）は、すべて第1水準に含まれ、人名用漢字別表（166字）のうち150字は第1水準に、16字は第2水準に含まれている。また、常用漢字表の本表には、先の1945字についての字体、音訓、語例などが総合的に示されている。さらに、国の国語施策の一環として、**現代仮名遣い**（昭和61年7月1日内閣告示第1号）や**改定送りがなの付け方**（昭和48年6月18日内閣告示第2号）などが公示されており、標準的な漢字使用並びに用例が示されている。その他、教育指導要領における996字、日本新聞協会新聞用語懇談会の1940字などは、特定の分野に於ける用字、用例を示している。一方、人名や地名をコンピュータ処理する金融業種や、電子製版が盛んな印刷業種からの強い要望により、1988年中に約6000字の新たな文字コードを定めるJIS第3水準制定の作業も順調に進んでいる。前記とは拮抗する内容だが、特定の分野では、このような要望も多い。

現在の日本語入力システムは、万人が利用する汎用性を備えている。その汎用性ゆえのトラブルも発生する。変換キーを押すと、ユーザーが一度も書いたことがないような漢字が“ポンポン”と出現する。読めなければ変更するが、なんとか読めれば、そのままにしておく。中には、読めなくても、そのまま放置(?)している猛者もいる。日本語入力システムの辞書の中には、明らかな登録ミス(誤字)も時々見られるが、それも見逃しがちだ。(ACE:手操る, HuTOP:竹内(シナイ)など)

公用文(特に、官公庁提出文)や学校教材などは、それぞれの分野における用字、用例を原則として守る必要がある。ところが、余程熱心に教育を受けたものでなければ、その全てに精通しているわけではなく、誤用字や誤用例を許してしまう可能性は高い。

パソコンにおける日本語入力システムは、その規模ならではの機動性を生かし、分野、用途別の表記制限を実施できれば、利用範囲が広がり、安心して使用できるシステムとなるだろう。

“常用漢字表準拠”や“学年別漢字入力システム”などといった単独システムであってもよいし、できれば、インストール時や起動中に**変換レベル**を指定できるものができれば、汎用性も確保できるだろう。現在の日本語入力システム技術を延長し、登録単語に表記別の**変換レベル記号**を記録し、送り仮名などの文法判断を表記別に限定すれば実現できる。例えば、小学校レベルの表記を選択した場合、それ以外の漢字を使用すると、自動的に“かな”が選択されたり、常用漢字表以外の漢字や表記を使用した場合は、表示色を変更し、特別な警告を発したりすることも可能である。医療用辞書などの分野別辞書のみが単独で販売されているが、システム自体にも辞書選択機能を持たせ、より広範な利用に耐えうるものを作らなければならない。いずれにしろ技術的な問題は少なく、実現性も高いと思われる。

6 おわりに

本稿では、かな漢字変換日本語入力システムの現状を述べ、さらに、今後の展望を考察してみ

た。1983年、MS-DOS Ver. 2.0の日本語メッセージを初めて見た時、“コンピュータらしくない”違和感を感じ、従来のスマートな英語のメッセージを懐かしくさえ思った。確か、そのころ作成したプログラムには、細やかな抵抗からか英語メッセージで統一しようと努力した形跡がある。その後、違和感もなくなり、逆に日本語の親切なメッセージがマンマシンインターフェースを高級なものに見せ、コンピュータの機能の高さを感じさせるようになった。昨今、手元に届くDMや明細書など、ほとんどが漢字かな文字になっている。カタカナ文字で打ち出された伝票を見て、コンピュータが打ち出したのだと改めて感心する時代も過ぎた。

コンピュータに“自然な日本語”が加速度的に導入され、その使用が必須のものになっている現在、日本語入力システムの操作の統一や辞書の共有が、ますます重要になってきている。

最後に本稿執筆にあたり多くの御助言を頂きました中日本自動車短期大学教授、福井 稔先生に紙面を借りて謝意を表します。

参 考 文 献

- (1) 三吉, 郡司, 白井, 橋田, 原田: 日本語の句構造文法—J P S G, コンピュータソフトウェア VOL.3 NO.4
- (2) 杉村良一: 日本語の待遇表現の解析と状況意味論, コンピュータソフトウェア VOL.3 NO.4
- (3) 箭内敏夫: 日本語フロントエンドプロセッサと国語表記原則, PROMPT 1988-2
- (4) ASCII 編集部: 日本語ワードプロセッサ—その歴史から明日まで, ASCII Vol6 #9 1982
- (5) 塩野 卓: かな漢字変換の変遷と今後の展望, ASCII Vol 9 #12 1985
- (6) 和多田作一郎: 人工知能のA B C, BLUE BACKS B735
- (7) 松山俊一: 万能でないフロントプロセッサ, NIKKEI PERSONAL COMPUTING 1986. 4.14

筆 者 記 注

- * 1 標準パターンとして持っているメモリ内容と実際の入力を比較照合させ、判定、認識を行うパターンマッチングに必要な時間軸正規化操作の一方法。
- * 2 本稿の執筆にあたり、新旧バージョンを含む次の13の日本語入力システムを使用した。
KNJDIC. SYS (83/09/20/, 86/09/30 : NEC), NECDIC. DRV (86/09/30, 87/10/23 : NEC), NECTIC. DRV, NECTIK2. DRV (87/10/23 : NEC), 松茸86 (Ver. 1.1, 1.5, 1.7 : ㈱管理工学研究所), 松茸V2 (Ver. 2.05 : 管理工学研究所), VJE- α (Ver. 1.1 : ㈱ボックス), VJE- Σ (Ver. 1.02, 1.10 : ㈱ボックス&㈱アスキー), ATOK 6 (Ver. 1.2 : ㈱ジャストシステム)
また、次の11種類の日本語入力システムの仕様を参考にした。
NECREN. DRV (88/10/13 : NEC), ATOK 5 (Ver. 1.1 : ㈱ジャストシステム), EG Bridge (Ver. 2.00 : ㈱エルゴソフト), FIXER 3 (Ver. 3.00 Rel. 1.1.0 : ソフトウェア・ソサエティ), 刀 (Ver. 1.04 : ㈱サムシング), MGR (Ver. 1.04 : ㈱リードレックス), SLAX (Ver. 1.1 : ㈱OAテック), VJE- β (Ver. 1.10 : ㈱ボックス&㈱アスキー), ACE (オーロラエース : ㈱大塚商会), WX (創文 α : エー・アイ・ソフト㈱), HuTOP (㈱ハドソン)
- * 3 通産省によるICOT(新世代コンピュータ技術開発機構: 淵 一博所長)の第5世代コンピュータ研究は有名で、並列処理による知識推論マシンの試作も完了している。
- * 4 1974年、MIT (Massachusetts Institute of Technology) のM. Minskyが発表した理論。人間の頭脳の中にある情報を、ある単位(フレーム)で構造化して、これを知識として人工知能の知識データベースに記憶させ、フレームの前後関係から推論を行う。

- * 5 1984年、東京大学 坂村 健により提唱されたコンピュータアーキテクチャ。リアルタイム性と共に共通の操作性を強調する純国産のコンピュータアーキテクチャである。産業用組み込みオペレーティングシステムのITRON (industrial-TRON)、ワークステーション用の強力な MMI をサポートする BTRON (business-TRON)、通信制御用のCTRON (communication-TRON)、ネットワーク分散制御システムである MTRON (macro-TRON) などの各 OS だけでなく、システムを効率良く実現する TRON チップと呼ばれる VLSI も開発している。
- * 6 1969年、米 AT&T、ベル研究所の Ken Thompson と Dennis Ritchie らが開発した汎用 TSS オペレーティングシステム。プログラム開発環境に優れ、そのソースプログラムの殆どが言語Cにより書かれているなど興味深い。現在、本家 AT&T 版の System V 以外にカリフォルニア大学バークレー校から配付されている4.2 BSD などの2つの流れがある。